## 11月16 《一些RBER-UBER图》

《2023. ALCod: Adaptive LDPC Coding for 3D NAND Flash Memory Using Inter-Layer RBER Variation》 抓像素点

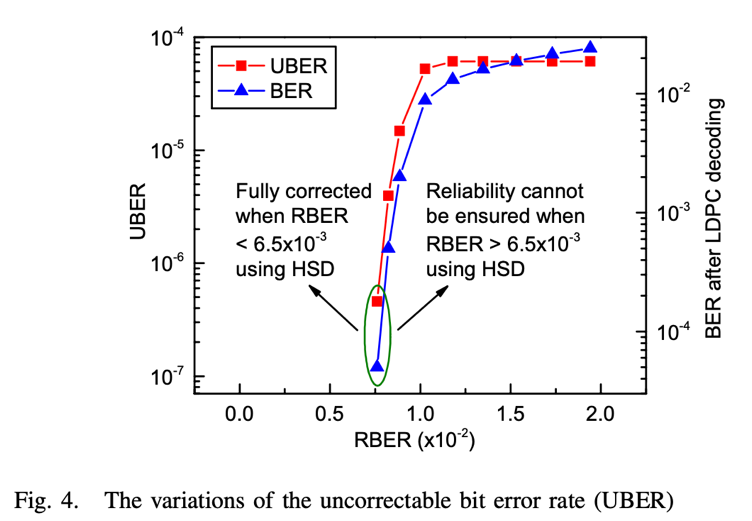
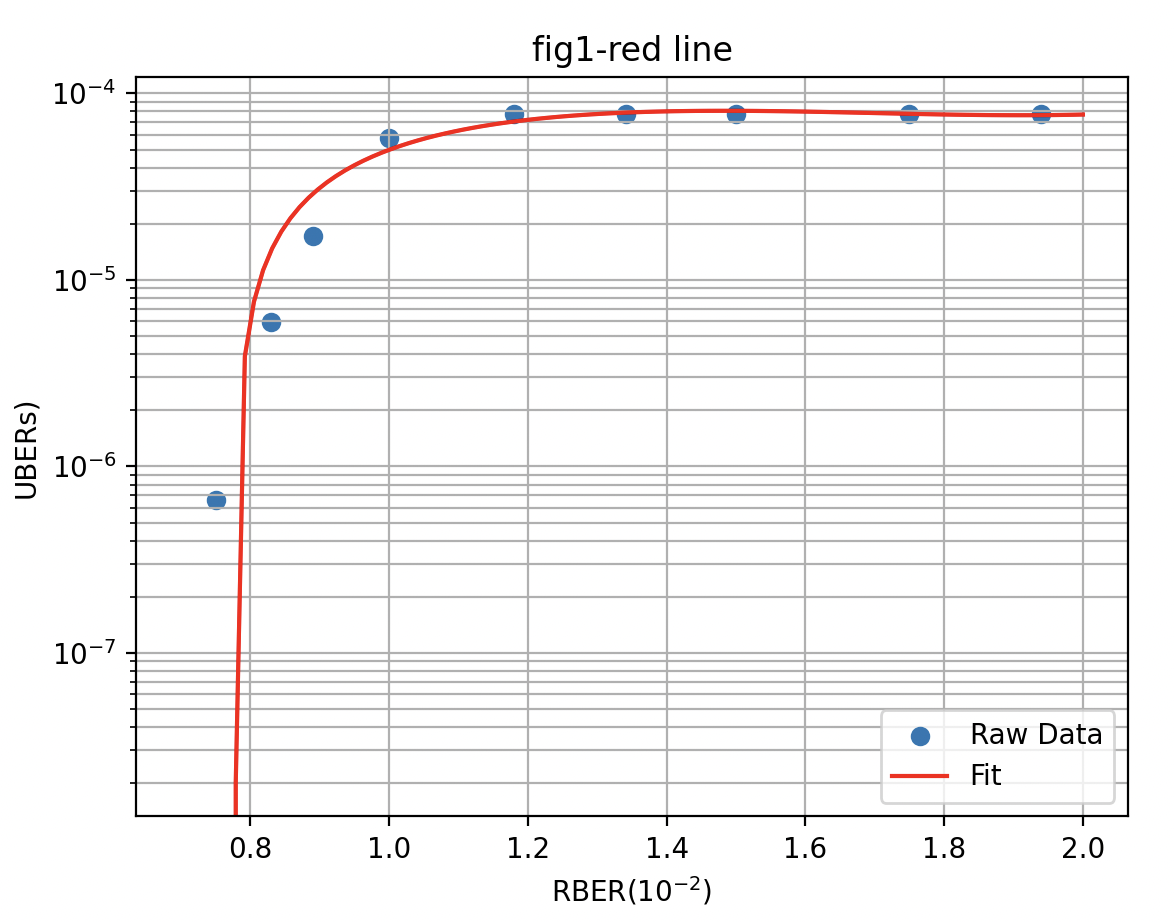


Fig1

红色曲线的拟合结果：

[a, b, c, d] = [ 7.33820523e-05, -1.29416212e-03,1.87978880e-03, -6.09156463e-04]

下图是拟合效果，蓝色的点是原数据点，红色的线是拟合的幂函数：



《<https://www.flashmemorysummit.com/English/Collaterals/Proceedings/2013/20130814_E22_YangJ.pdf>》 2013

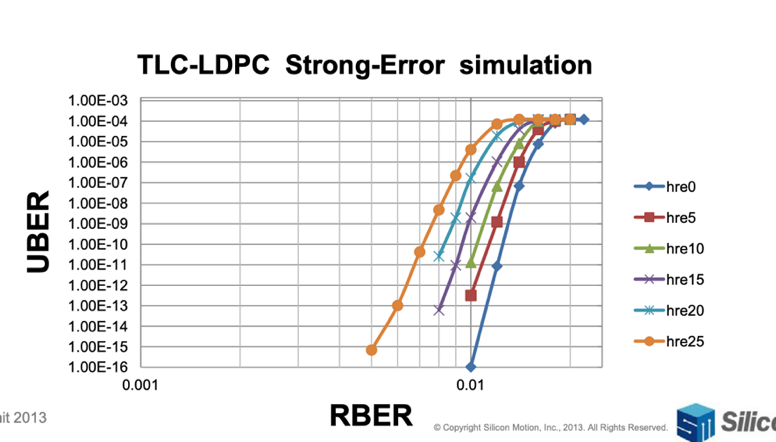


Fig 2

《A 38.64-Gb/s Large-CPM 2-KB LDPC Decoder Implementation for NAND Flash Memories》

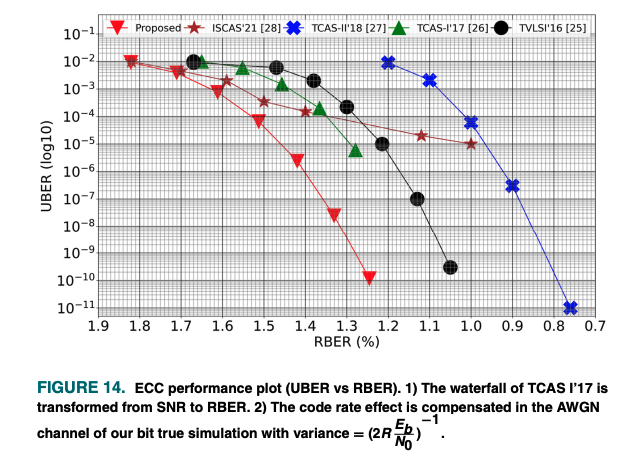
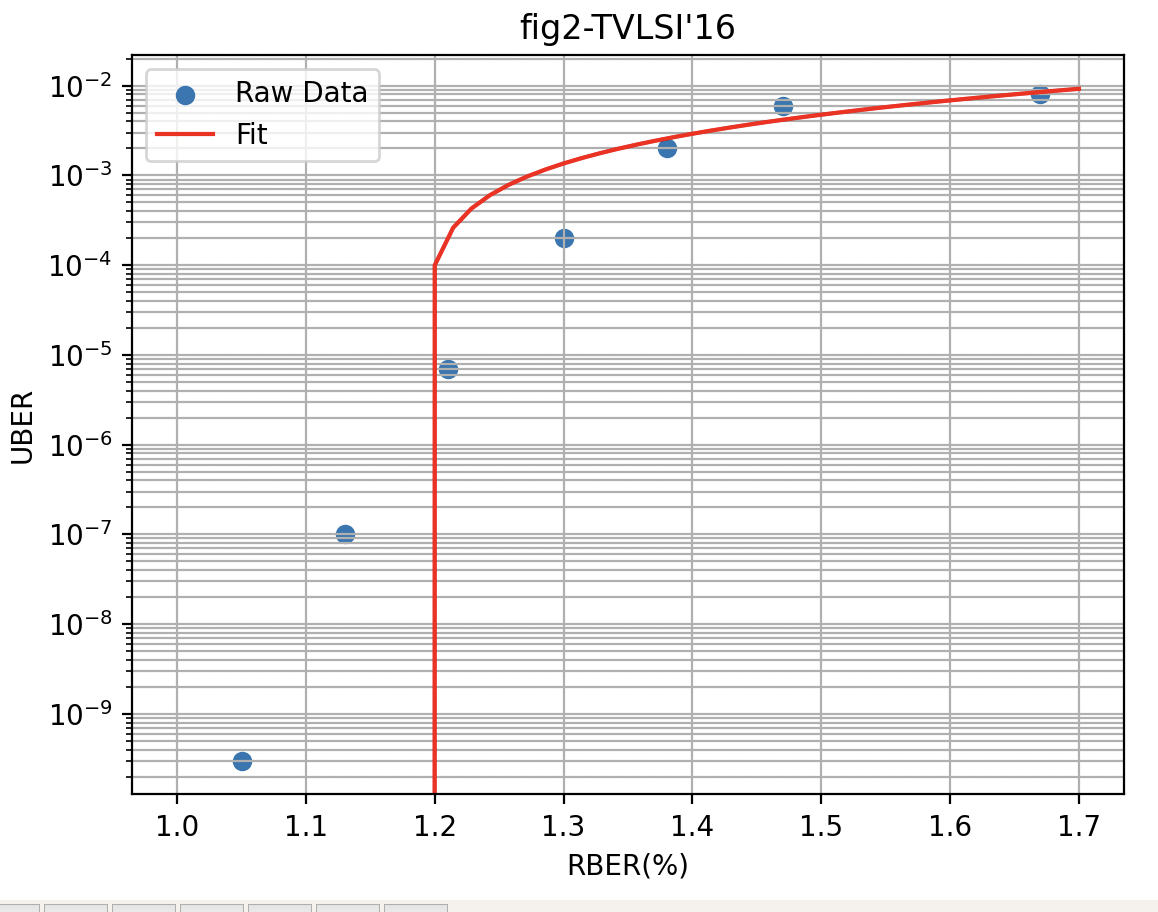


Fig 3

这两张图的拟合效果不好，例如下面是拟合上图黑线的结果，其他的线拟合结果和他差不多，都是左边的点没法拟合。



《A Energy-optimized (37840, 34320) Symmetric BC-BCH Decoder for Healthy Mobile Storages》

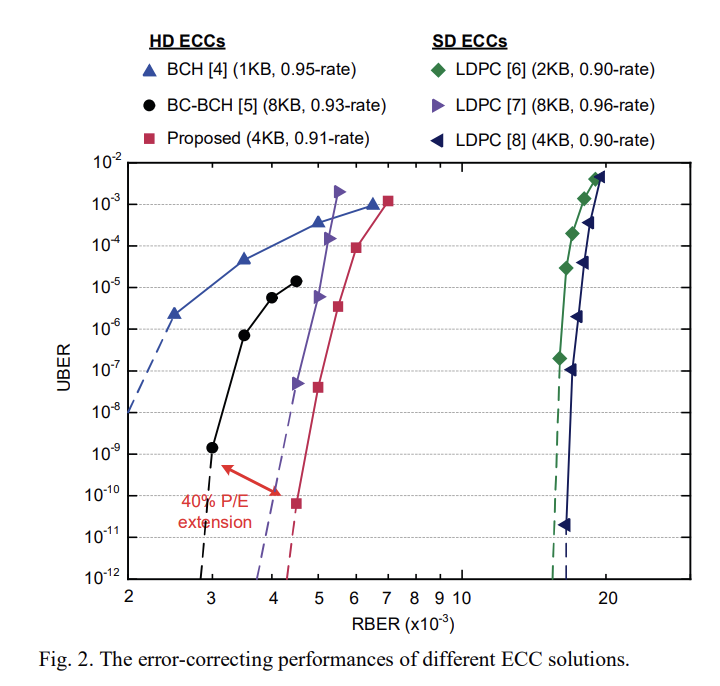
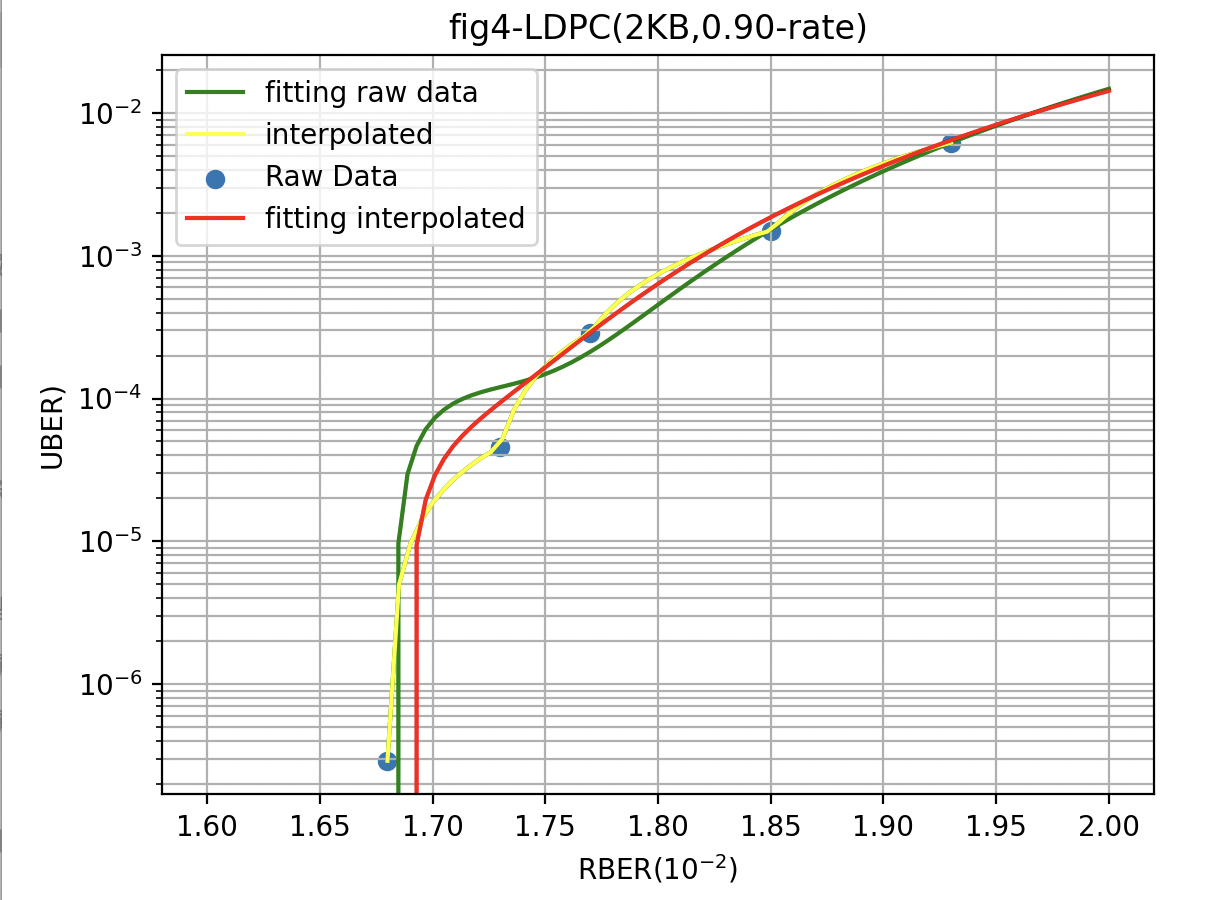


Fig 4

这张图尝试拟合了上图中的绿线（2KB LDPC）。在下图中，蓝色圆点表示原数据点，绿线是基于原数据点拟合点幂函数；黄线表示用线性插值做的图，红线是基于线性插值的点拟合点幂函数。



Fitting raw data: [a, b, c, d] = [ 0.56438078, -2.89666958, 4.95777382, -2.82961658]

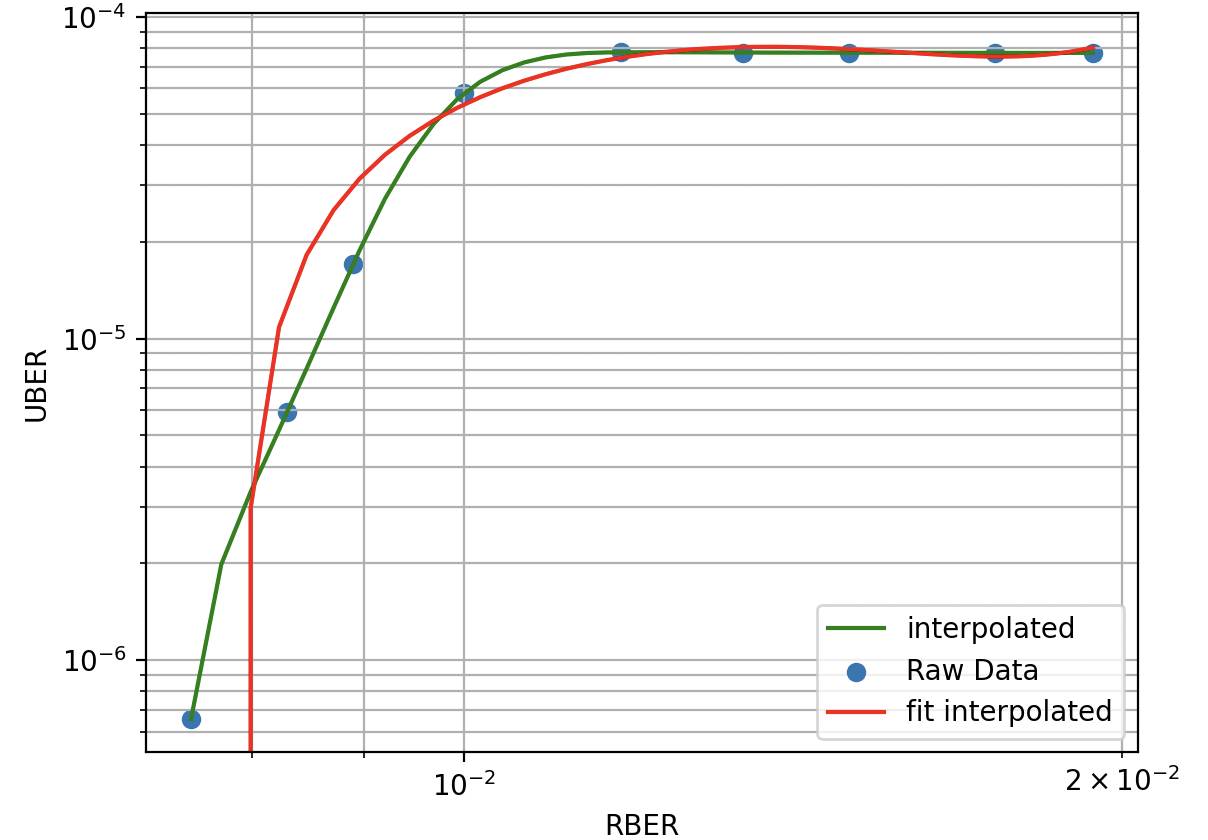
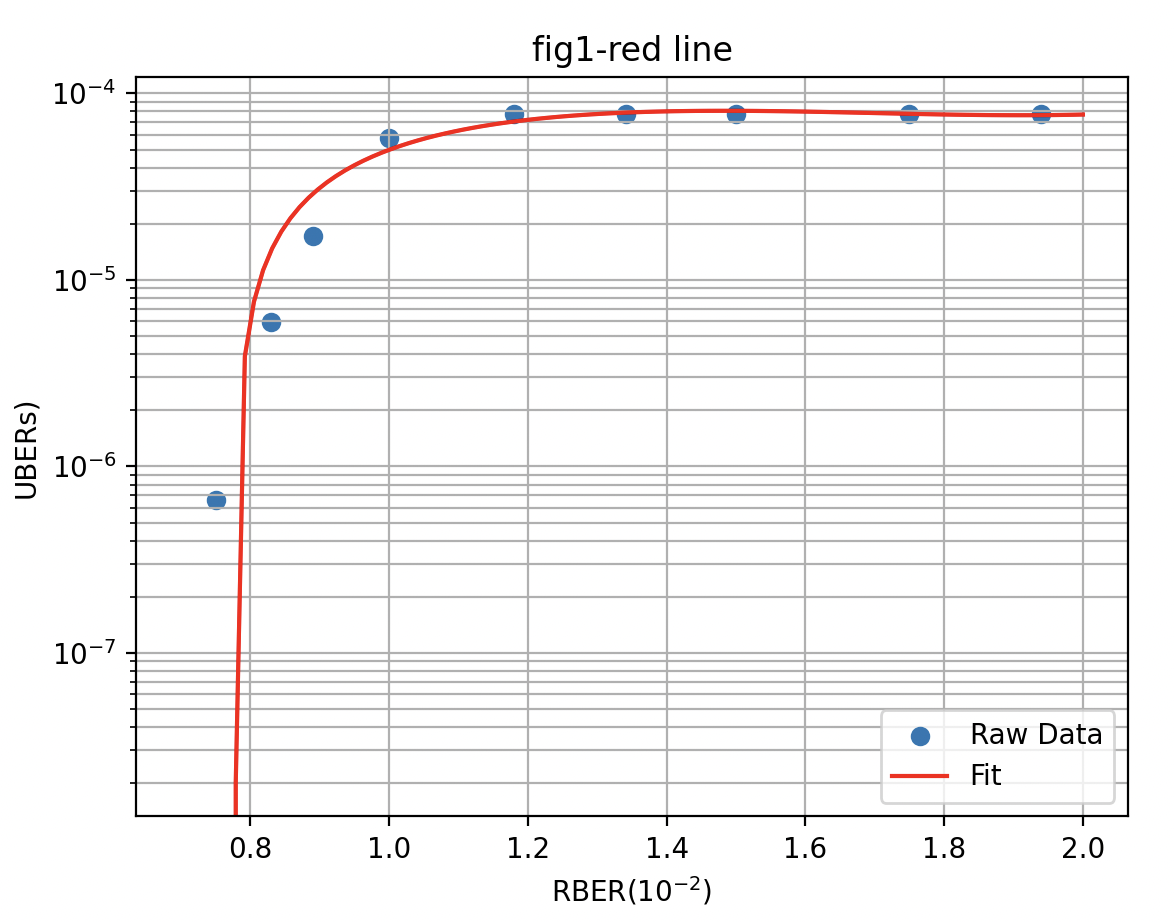
Fitting interpolated: [a, b, c, d] = [ 0.72151759, -3.74249357, 6.47181204, -3.7310196 ]

## 11月17 三次样条插值法

现尝试使用三次样条插值去取拟合函数，但是三次样条插值法和线性插值一样，都是用来增加样本点的方法，无法直接得到拟合曲线的函数关系式。现有下面两种方式得到函数关系式：

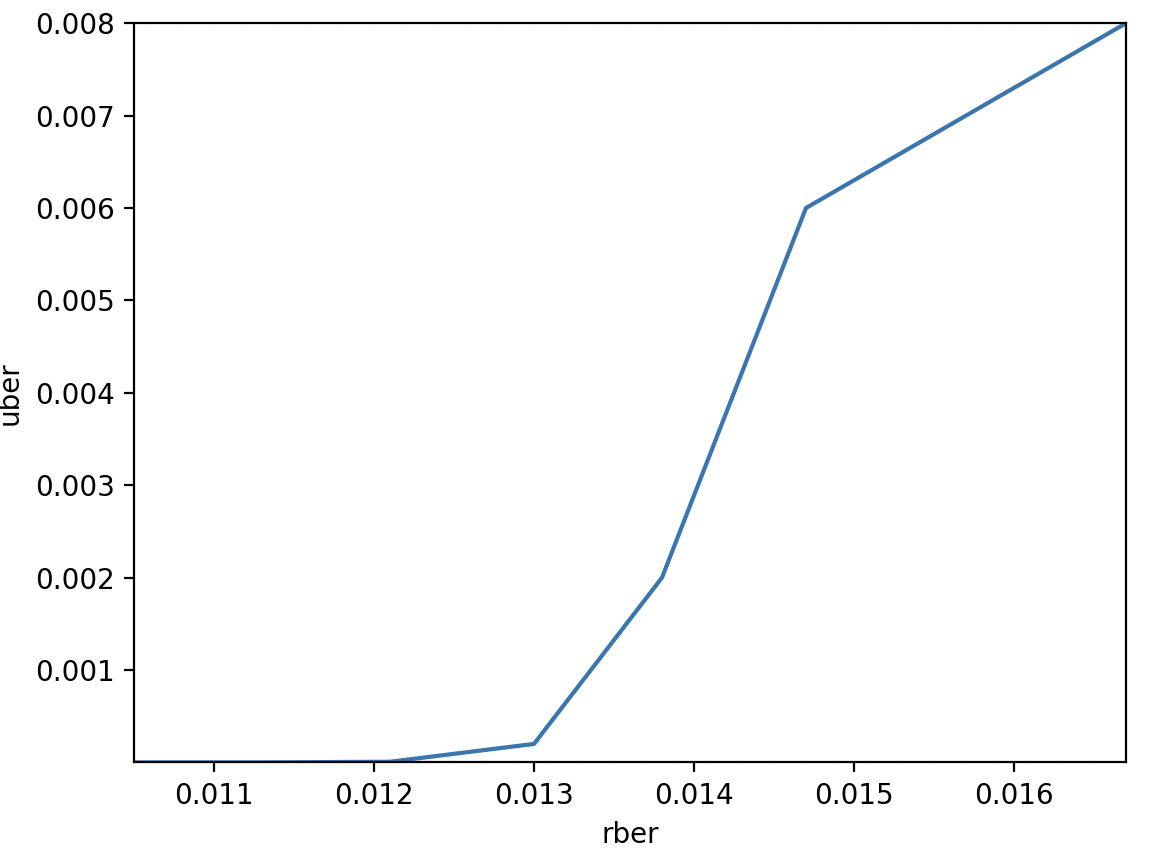
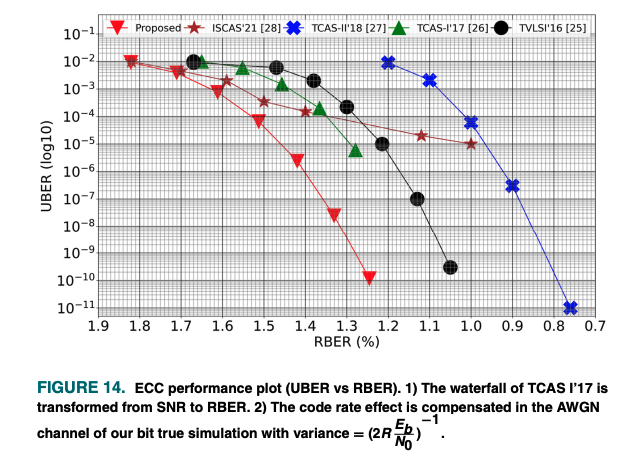
1. 通过插值点进行计算：可以使用插值点和对应的函数值来计算拟合曲线的函数方程。您可以选择一些特定的x值，然后使用cs(x)来获取对应的拟合函数值。这样，您可以得到一系列(x, f(x))的点，然后使用这些点来拟合一个函数方程。常见的方法包括用多项式拟合或使用其他曲线拟合方法，如最小二乘法。
2. 输出插值多项式：CubicSpline对象提供了一个属性cs.c，它返回在每个插值段上的插值多项式的系数。这些系数可以用来表示每个插值段上的多项式函数。您可以使用这些系数来构建整个拟合曲线的函数方程。请注意，这种方法只给出了拟合曲线在每个插值段上的多项式表达式，并不提供整个曲线的连续函数表达。

方案2)我试了，和直接用原数据点拟合的函数相比，只是系数发生了点变化，没有提升拟合效果，下面左图是直接拟合的，右图是基于三次样条插值得到的点拟合的。



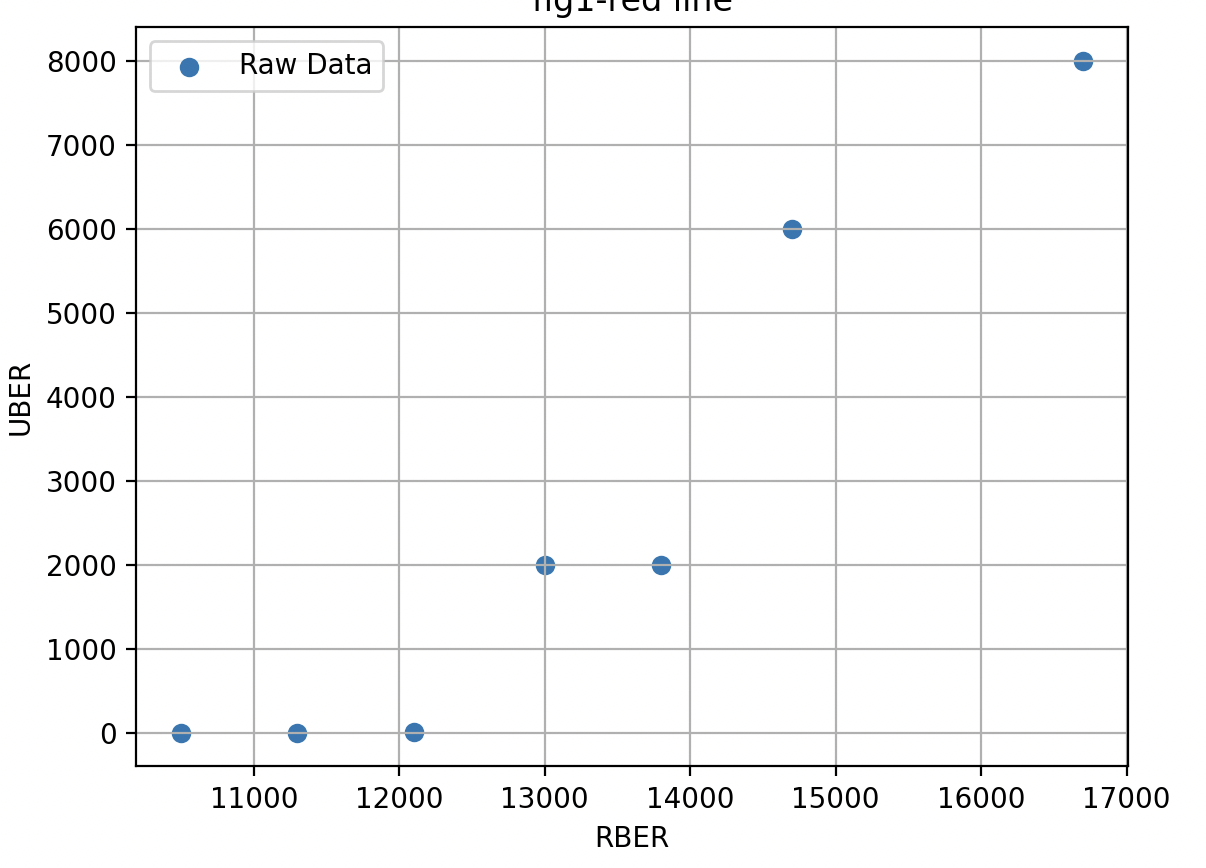
下面使用放大法和取对数法进行拟合。

Fig3 中的黑色圆圈线，看着很光滑，其实原数据并不光滑，下图左边是论文中的图，右图是实际坐标下的图

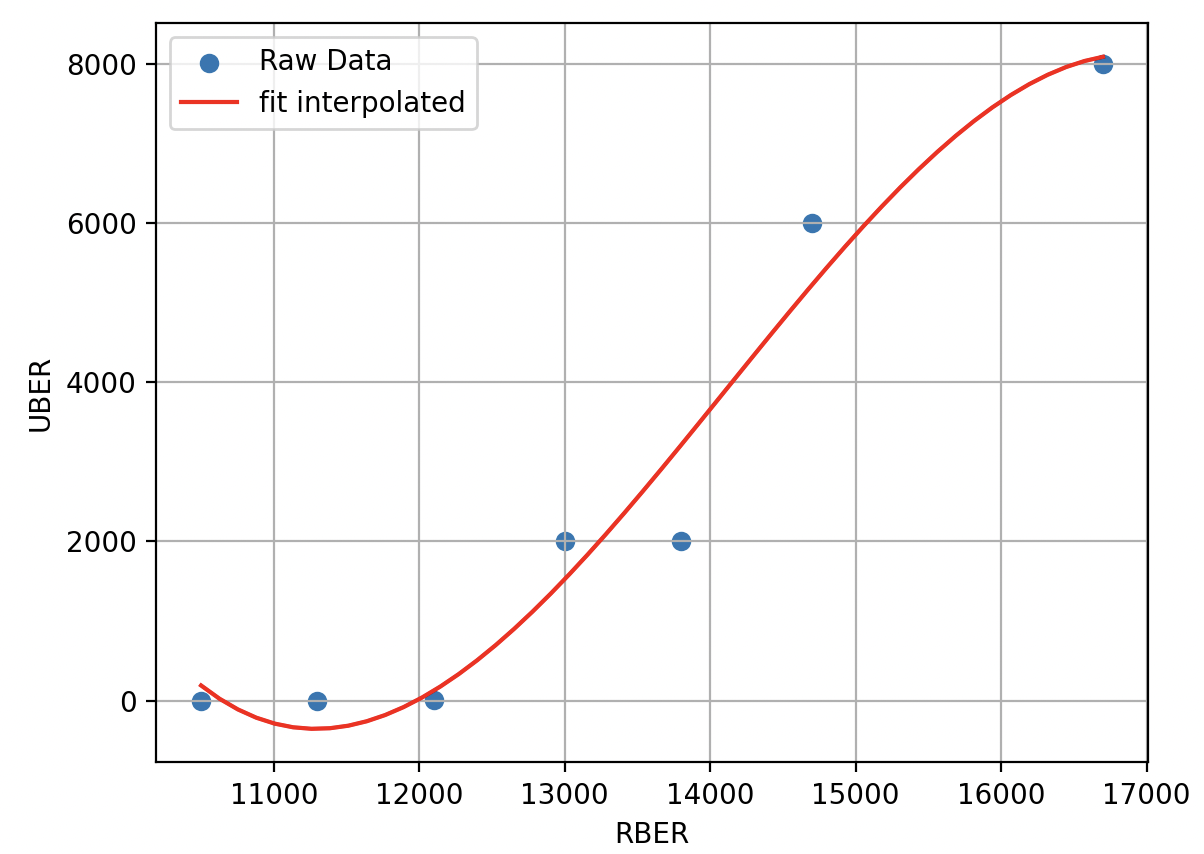


## 放大法

现尝试使用放大法进行拟合。把上面的右图x, y都放大倍：

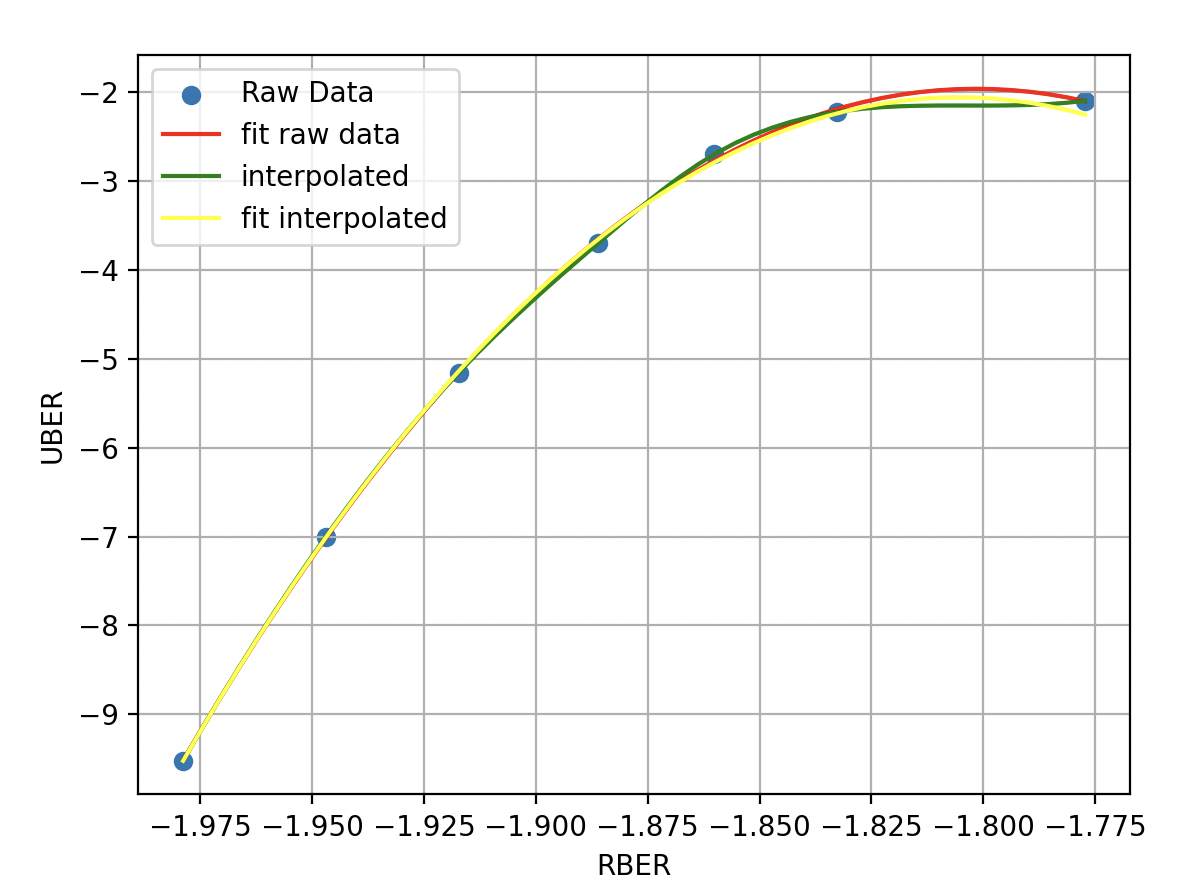


用三次幂函数拟合上图，可以看到拟合效果也不好：



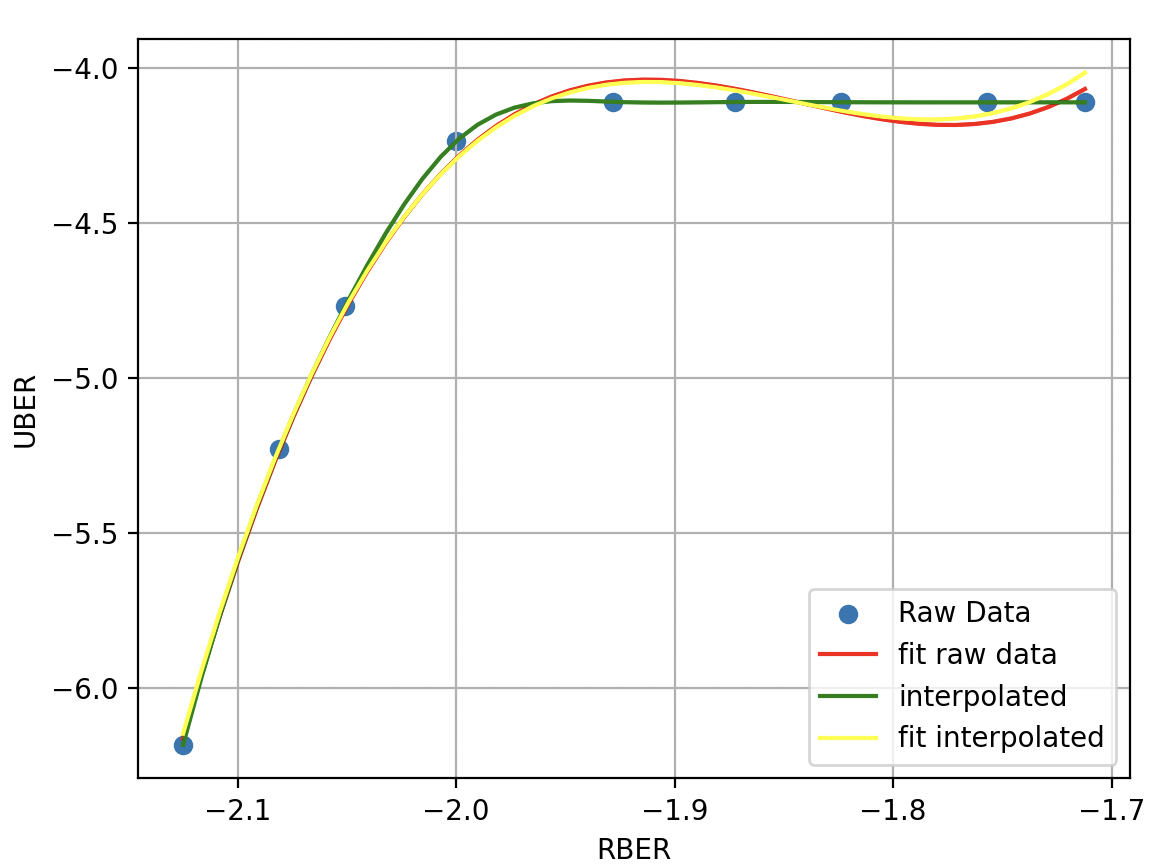
## 取对数法

然后尝试使用取对数方式，x, y都对10 取对数，再次拟合：

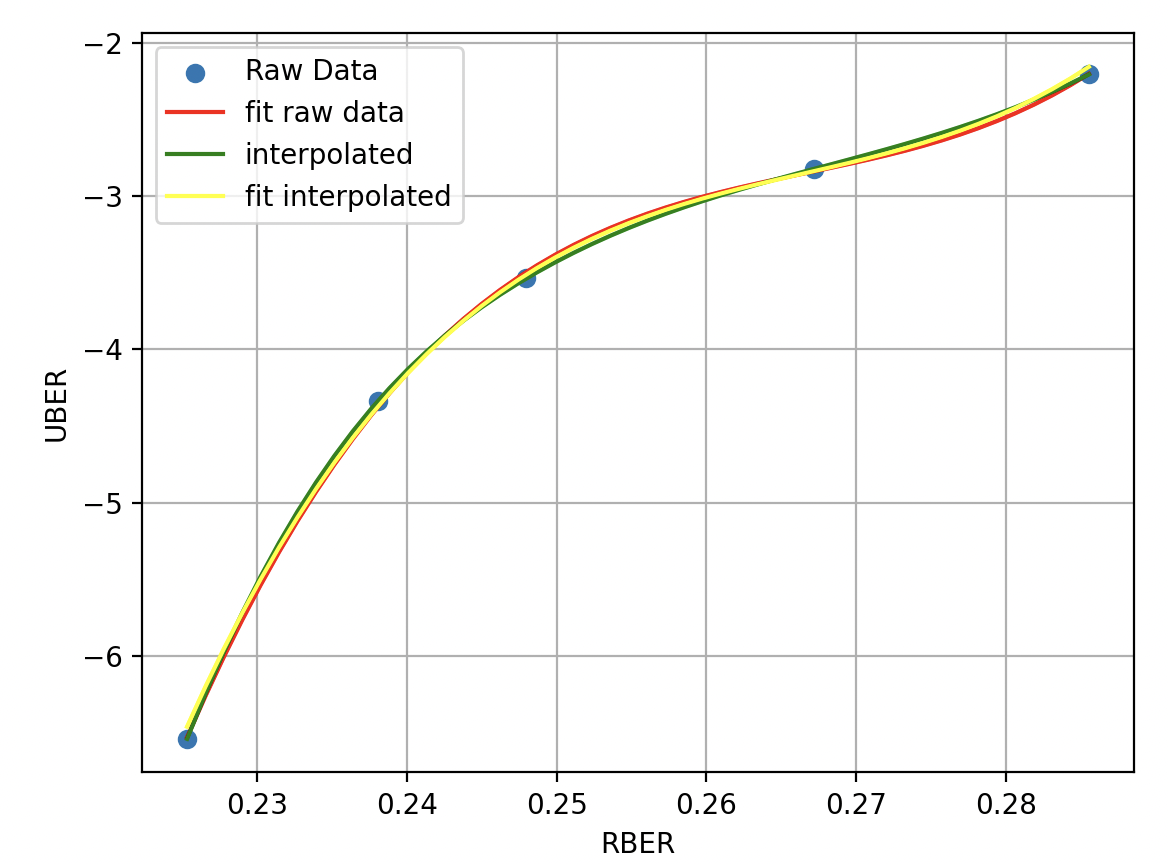


过拟合了吗？

同样的方法去拟合一下fig 1：



拟合一下fig4:



结果都不错，由此可以得出结论：

使用先取对数，再幂函数拟合的方式，可以比较好的拟合出rber-uber的函数关系。例如针对fig4的拟合过程，拟合的函数为：

其中：

*Fit raw data:*

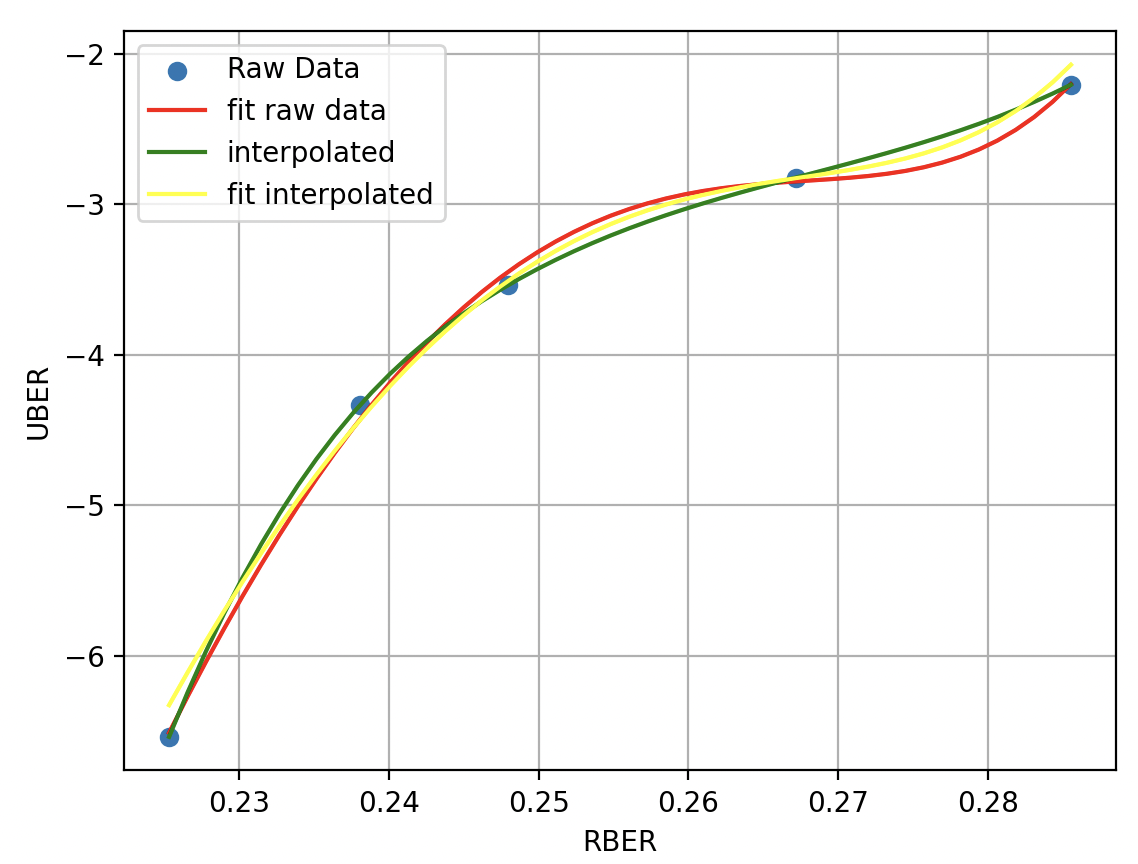
*[a, b, c, d] =* [39712.27295069, -31786.14656006, 8500.99286288, -762.49571563]

*Fit interpolated:*

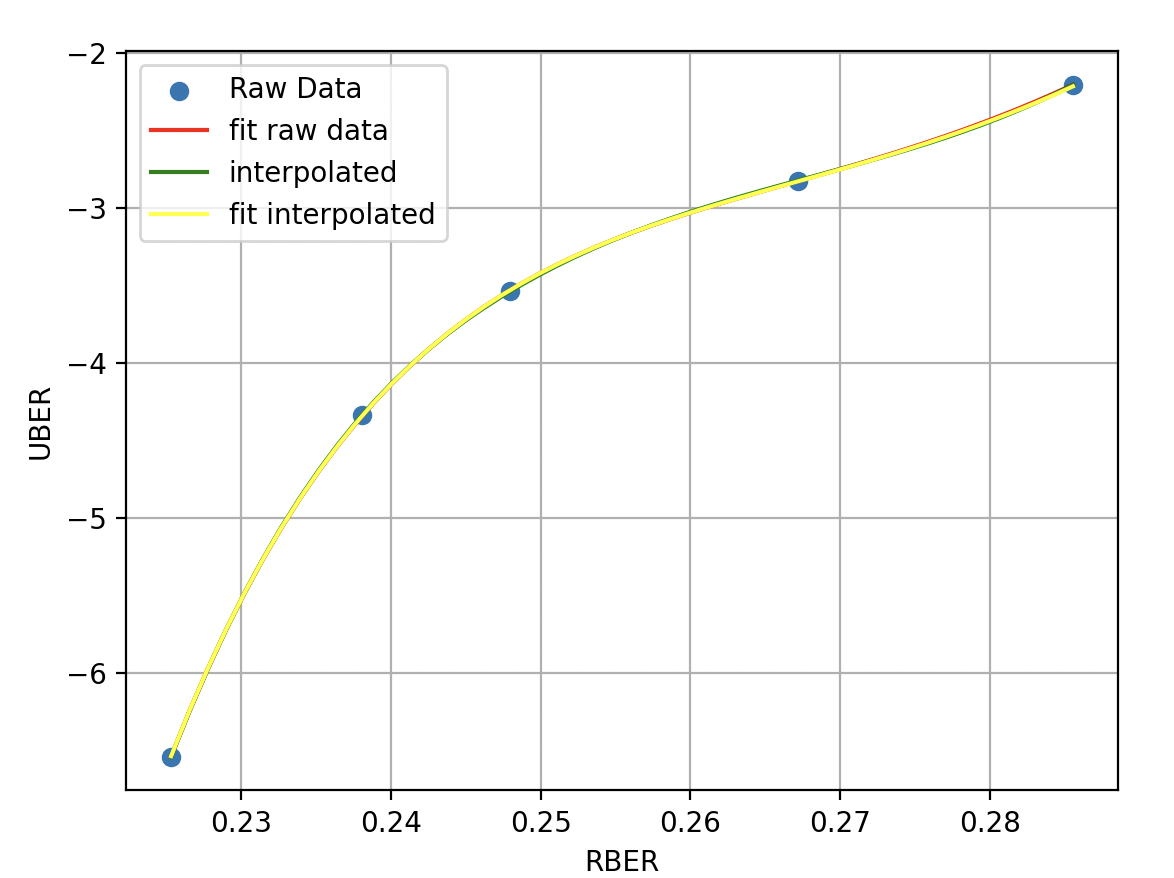
*[a, b, c, d] =*[ 38487.55733562, -30766.09216987, 8220.32831514, -736.964264]

现在考察一下，次幂对拟合结果的影响：

上面的的次幂是(3, 2, 1)，现增大次幂为(9, 7, 4)，拟合结果如下图所示



减小次幂为(-1, -2, -3)，不得不怀疑过拟合？



并且多次更改次幂，实验结果发现，次幂的变化为通过系数来修正，但次幂的绝对值越大，拟合误差越大。

现考虑多项式的项数的影响，增加项，新增项的影响会被其他项的系数修正，但整体稍微更准确点，由于数据和时间原因，暂不深究具体影响，先到这。